Crash Course on Protégé

Petr Křemen

v1.1

1 Introduction

Protégé is an ontology editor. Its earlier versions (Protégé 3 and lower) were used for creating ontologies in frame-based formalisms. The third and fourth major versions of Protégé allows creating OWL (Protege 3) and OWL 2 (Protege 4), respectively. The software is open-source and downloadable from `http://protege.stanford.edu`.

1.1 Web Ontology Language (OWL) in a Few Sentences

Description logics (DLs) provide logical and reasoning calculus to OWL ontologies. Yet, in order to be practical, OWL is much richer.

- OWL contains, in addition to *object properties* (corresponding to description logics roles), also *data properties* that connect individuals to basic data (like strings, or integers) and allow reasoning about them, and *annotation properties* that connect any **entity** (class, object property, data property, annotation property, data type, individual) to a any other entity or basic data and *without the possibility* to reason about them.

- OWL identifies all entities with URIs in order to be addressable in the semantic web space.

There is much more about OWL. Please look at `http://www.w3.org/TR/owl2-primer`.

1.2 Correspondence between DLs and OWL

There is different terminology, as well as different syntax, between DLs and OWL.

1.2.1 Terminology

As for the terminology, here is the correspondence between OWL terms and DL terms:

description logics	OWL
concept	**class**
role	**object property**
constant/individual	**individual**
theory	**ontology**
axiom	**axiom**

1.2.2 Syntax

As for the syntax, OWL uses many serializations and syntaxes. E.g. the OWL file you download in Section 3, is RDF/XML serialization of the OWL ontology. Protégé parses various serializations and shows up the OWL ontology in so called Manchester Syntax (see `http://www.w3.org/TR/owl2-manchester-syntax`). The Manchester syntax tries to keep the axioms readable for non-experts in description logics. Here are the most important parts of the translation:

description logics syntax	Manchester syntax (OWL in Protégé)
$C_1 \sqsubseteq C_2$	C_1 **SubClassOf** C_2
$C_1 \equiv C_2$	C_1 **EquivalentTo** C_2
$C_1 \sqsubseteq \neg C_2$	C_1 **DisjointWith** C_2
$R_1 \sqsubseteq R_2$	R_1 **SubPropertyOf** R_2
$\neg C$	**not** C
$C_1 \sqcup C_2$	C_1 **or** C_2
$C_1 \sqcap C_2$	C_1 **and** C_2
$\exists R \cdot C$	R **some** C
$\forall R \cdot C$	R **only** C
$\exists R \cdot \{i\}$	R **value** $\{i\}$
$(\geq\ 2\, R\, C)$	R **min** 2 C
$(\leq\ 2\, R\, C)$	R **max** 2 C
R^-	**inverse** R

2 Installation

- Download and install Protégé 4.3 from `http://protege.stanford.edu/download/registered.html`. You need to use Java SE 6 or 7.
- Install at least the following Protégé plugins – `File` → `Check for plugins...`
  - Pellet Reasoner Plug-in (more info at `http://clarkparsia.com/pellet/`)
  - OWLDiff Protégé Plugin (more info at `https://kbss.felk.cvut.cz/web/portal/owldiff`)

3 Short tutorial

- Download the Pizza Ontology from the following URL: `http://130.88.198.11/co-ode-files/ontologies/pizza.owl`.
- Run Protégé and open the Pizza ontology.
- You will see an application window with several tabs – become familiar namely with the Entities, Classes, Object Properties and Data Properties window:

  Active Ontology – here, you can find the information about ontology metrics (number of classes, object properties, data properties, axioms) and the corresponding description logic expressiveness, the ontology uses (in the case of pizza.owl you should see $\mathcal{SHOIN}$).

  Entities/Classes – here, you will probably spend most time when working with Protégé. Left, you can see a class hierarchy (subsumption axioms rendered as a tree) before classification (Asserted class hierarchy) and after classification (Inferred class hierarchy). In the central part, there are details about the class selected in the left tree – specific anotations, axioms in which this class participates.

  Object/Data Properties/Individuals – analogous to Classes view.

  OWLViz – shows simple and intuitive visualization of TBox.

  DL Query – allows posing simple ontological queries, e.g. "Find all subclasses/instances/... of the given class".

- Try classifying the ontology – go to the menu "Reasoner", choose Pellet, or FaCT++, or HermiT and select the "Start reasoner" menu item. Look what are the differences before (Asserted class hierarchy) and after (Inferred class hierarchy) the classification.

4 Other Relevant Resources

- Ontology Modeling Tutorial with Protégé:
  `http://protege.stanford.edu/publications/ontology_development/ontology101.pdf`
- Getting Started with Protégé 4:
  `http://protegewiki.stanford.edu/index.php/Protege4GettingStarted`
- A Practical Guide to Building OWL Ontologies Using Protégé 4 and CO-ODE Tools, Ed. 1.3,
  `http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4\_v1\_3.pdf`
- A list of ontology authoring and management tools,
  `http://www.mkbergman.com/862/the-sweet-compendium-of-ontology-building-tools`.